

# Identificação Inteligente de Linha de Visada em Redes LoRaWAN

Lucas L. de Oliveira, Leonardo A. de Oliveira, Edelberto F. Silva e Álvaro A. M. de Medeiros

**Resumo**—Um problema significativo nos sistemas de internet das coisas é a flutuação de um enlace entre os estados de linha de visada e sem linha de visada. A discriminação entre tais estados é importante para uma variedade de propósitos, incluindo modelagem de canal, dimensionamento de transmissores e agendamento de transmissões. Este artigo implementa técnicas de aprendizado de máquina para classificação de linha de visada, a partir de dados coletados no campus da Universidade Federal de Juiz de Fora e os resultados indicam melhor desempenho para Random Forest e K-Nearest Neighbors nesse cenário.

**Palavras-Chave**—IoT, LoRaWAN, LOS, NLOS, Aprendizado de máquina.

## I. INTRODUÇÃO

Com base nos recentes avanços tecnológicos, o número de aplicações de Internet das Coisas (IoT) cresceu significativamente nos últimos anos [1]. A maioria das aplicações IoT não requer altas taxas de dados, portanto neste cenário, tecnologias *Low-Power Wide Area Network* (LPWAN) surgem como uma solução que permite comunicação de longo alcance e baixo consumo de energia.

Dentre as principais tecnologias LPWAN utilizadas em IoT, temos a *Long Range Wide Area Network* (LoRaWAN), que se baseia na técnica de camada física LoRa para fornecer um sinal que sobressaia sobre ruídos e interferências [2].

A implementação de redes LoRaWAN tem se destacado como boa opção para cenários de *smart cities* [3], nos quais problemas como mobilidade urbana são abordados [4]. Congestionamentos das vias urbanas nas áreas de maior circulação de veículos são problemas em muitas cidades brasileiras, bem como na cidade de Juiz de Fora (MG). Especialmente no campus da Universidade Federal de Juiz de Fora (UFJF), há um expressivo número de estudantes circulando durante todo o dia, além de o campus estar localizado entre os bairros Aeroporto e São Pedro, sendo a principal conexão entre o centro da cidade e esses bairros. Com isso, o deslocamento de alunos e servidores dentro do extenso (134,68 hectares) e acidentado (até quase 100m de diferença de altitude) campus da universidade é dificultado, principalmente pelo fato dos ônibus circulares internos não possuírem nenhum tipo de monitoramento de frota, provocando incertezas para os usuários quanto à localização dos mesmos.

Uma possível solução é o uso de dispositivos finais (EDs) LoRaWAN para transmissão de informação sobre posicio-

namento de ônibus circulares e situação de tráfego dentro do campus [5]. Os EDs transmitem para um *gateway* (GW) dentro do campus que disponibiliza a informação para servidores e aplicativos em terminais celulares ou monitores em paradas. No entanto, desconhecer o cenário em que uma rede LoRaWAN atua pode fazer com que o desempenho dos dispositivos presentes na rede não seja satisfatório, uma vez que o dimensionamento inadequado pode resultar em menor taxa de transmissão, redução de cobertura ou aumento desnecessário do consumo de energia [6]. Nesse contexto, surge a necessidade de conhecer características do meio em que a rede atuará, como a presença de enlaces com linha de visada ou *Line-Of-Sight* (LOS) e sem linha de visada ou *Non-Line-Of-Sight* (NLOS), a fim de ajustar parâmetros de transmissão de forma a maximizar o desempenho da rede e reduzir o consumo de recursos computacionais.

Este trabalho tem o objetivo de avaliar o desempenho de técnicas clássicas de aprendizado de máquina, mais especificamente *K-Nearest Neighbors* (KNN), *Multi Layer Perceptron* (MLP), *Support Vector Machine* (SVM) e *Random Forest* (RF), para classificar os dados de medição de potência de uma rede LoRaWAN de forma que sejam identificados como sinais com ou sem linha de visada. Para isso, é realizada uma campanha de medição na região do campus da UFJF, na qual são coletadas informações necessárias a fim de verificar os pontos LOS e NLOS presentes nessa região. A escolha das técnicas citadas se dá pela característica não-linear do problema [16], já que as variáveis para classificação são não lineares, como latitude, longitude, potência de recepção dada pelo parâmetro *Received Signal Strength Indicator* (RSSI), entre outras. Além disso, este artigo disponibiliza os dados coletados através da campanha, bem como as rotinas utilizadas para processamento e avaliação de desempenho descritos em [17].

### A. Trabalhos relacionados

Diversas soluções de classificação LOS/NLOS são encontradas na literatura com aplicações em localização [7], [8], [9], orientação do dispositivo [10] e sensoriamento [11] em diferentes redes de comunicação, tais como IoT *outdoor* [9] e *indoor* [7], veiculares [11] e industriais [12]. Entretanto, no melhor do conhecimento dos autores, há uma lacuna na literatura neste tipo de classificação para redes LoRaWAN em aplicações IoT *outdoor*. A caracterização de canal com classificação LOS/NLOS auxilia na otimização de parâmetros das redes LoRaWAN, propiciando um menor tempo de entrega de pacotes (*air time*), e no desenvolvimento de dispositivos de menor consumo energético, nos quais o amplificador de radiofrequência pode ser acionado de forma mais eficiente [13]. Ademais, a classificação LOS/NLOS auxilia na caracterização

Lucas L. de Oliveira e Álvaro A. M. de Medeiros são do Programa de Pós Graduação em Engenharia Elétrica da Universidade Federal de Juiz de Fora, Juiz de Fora-MG, Brasil, e-mail: {lima.lucas, alvaro}@engenharia.ufjf.br; Leonardo A. de Oliveira e Edelberto F. Silva são do Programa de Pós Graduação em Ciência da Computação da Universidade Federal de Juiz de Fora, Juiz de Fora-MG, Brasil, email: {leonardo.azalim,edelberto}@ice.ufjf.br. Esse trabalho foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Código de Financiamento 001.

da perda de percurso [14] e nos algoritmos de localização e rastreamento de EDs [15], complementando as informações obtidas por sistemas de posicionamento via satélite.

A contribuição desse trabalho consiste na classificação LOS/NLOS com técnicas de aprendizado de máquina para ambientes *outdoor*, tal como nos trabalhos anteriores [7] e [8], ambos para ambientes *indoor*.

O restante do artigo está organizado da seguinte forma: Seção II detalha a campanha de medição. Já a Seção III apresenta as técnicas de classificação e as métricas de desempenho utilizadas, enquanto a Seção IV mostra os resultados. As considerações finais estão contidas na Seção V.

## II. CAMPANHA DE MEDIÇÃO

Buscando a caracterização do canal com a identificação de visada, a campanha de medição foi realizada ao longo do campus da UFJF percorrendo os pontos de maior circulação dos ônibus circulares. Visando cobrir uma região de relevo acidentado e composta de prédios com diferentes alturas, uma antena que opera na faixa de 915MHz foi instalada no topo do prédio Itamar Franco da Faculdade de Engenharia, visto que este ponto está localizado em uma região de maior altitude e possui visada direta para alguns prédios da universidade. O GW foi disposto no mesmo prédio para facilitar a comunicação e, completando a rede LoRaWAN, um ED foi desenvolvido para capturar os dados em movimento pelo campus da UFJF. Por fim, o GW foi conectado à infraestrutura da *The Things Network* (TTN) e o armazenamento de dados foi realizado a cada 8 segundos. A Tabela I detalha as especificações técnicas da antena.

### A. Configuração dos equipamentos

Para redução de custos, o GW e o ED foram desenvolvidos a partir de um kit de desenvolvimento LoRaWAN composto de módulos MTAC (GW) [18] e mDot (ED) [19] da Multitech. Seguindo a mesma linha, optou-se por utilizar um ED mais simples, que não possui função de geolocalização nativa, conectado a um celular responsável por capturar sinais de GPS. Além da economia financeira, a conexão a um celular permitiu maior precisão de GPS se comparado com outros modelos de ED que possuem a função de geolocalização nativa disponíveis no mercado. As Tabelas II e III detalham os dispositivos utilizados como GW e ED, respectivamente.

A comunicação com o celular é realizada através de um cabo USB e auxílio do módulo *Android Debug Bridge* (ADB). Com isso, o celular captura os dados de altitude, latitude, longitude, número de satélites utilizados e precisão do GPS. Tais informações são enviadas para um computador que também está conectado ao módulo mDot através de um cabo USB, permitindo que elas sejam salvas no disco local em tempo real. Enquanto o celular é responsável por capturar as informações descritas, o módulo mDot foca apenas na coleta de valores de RSSI medidos tanto do lado do ED quanto do GW.

Dois *scripts* relacionados a TTN são executados em sequência, o primeiro para conectar a TTN ao ED através do protocolo *Message Queue Telemetry Transport* (MQTT) e o segundo para consultar o registro dos pacotes armazenados [20]. Como cada pacote possui seu identificador, facilmente pode-se integrar as

TABELA I  
CONFIGURAÇÕES DA ANTENA.

Parâmetro	Valor
Tipo	Polidesign PLD COL890-9605
Material Externo	Alumínio
Material Interno	Fibra de Vidro
Comprimento	1,80m
Diâmetro	44mm
Diretividade	Omnidirecional
Ganho	9dBi
Impedância	50Ω

TABELA II  
CONFIGURAÇÕES DO *Gateway*.

Parâmetro	Valor
Processador	Raspberry Pi 3B+
RAM	1GB
Armazenamento	Samsung SD Card EVO Plus 64GB (130MB/s)
Fonte de alimentação	12V /3A
Módulo LoRa	MTAC-LORA 94557300LF4
Ganho do Módulo	3dBi
Impedância	50Ω

TABELA III  
CONFIGURAÇÕES DO *End Device*.

Parâmetro	Valor
Tipo	Multitech mDot 9153
Modo de operação	<i>Over-the-Air</i>
Largura de banda	125kHz
Fator de espalhamento	12
Potência de transmissão	30dB
Ganho da Antena	5dBi
ADR	Desligado
Frequência	915MHz
Sensibilidade	RSSI ∈ [-140, 0]dB

informações referentes à localização junto as demais armazenadas pela TTN. Os parâmetros coletados são: *Time* (UTC -3), *GPS Time* (UTC +0), *id*, Latitude (graus), Longitude (graus), Altitude (m), *GPS Precision*, *# Satellites*, ED RSSI (dB), GW RSSI (dB).

Os parâmetros *Time* e *GPS Time* são marcadores temporais, sendo o horário da coleta da amostra de acordo com o fuso horário local (UTC-3) e com o fuso horário do meridiano de Greenwich (UTC+0), respectivamente, enquanto o parâmetro *id* indica o número da amostra do conjunto de dados. Já Latitude, Longitude e Altitude são parâmetros de localização obtidos pelo GPS presente na configuração montada, *GPS Precision* representa a precisão do GPS, em que o menor valor indica mais precisão e o maior valor menos precisão, sendo sua variação de 1 a 5 e *# Satellites* indica o número de satélites utilizados para obter essas informações. Por fim, ED RSSI e GW RSSI são indicadores de potência do sinal recebido para o *End Device* e *Gateway*, respectivamente.

### B. Coleta de dados

Com a rede configurada, a campanha foi executada seguindo o trajeto dos ônibus circulares no campus da UFJF e foi dividida em dois dias de medição. No primeiro dia, o ED é carregado por um indivíduo a pé, posicionado a uma altura aproximada de 1m em relação ao solo e percorrendo os prédios de Engenharia, Restaurante Universitário, Artes, Ciências Exatas, Biologia, Letras, Reitoria, Direito, Jornalismo e Ciências Humanas com uma velocidade média aproximada de 6km/h.

Enquanto isso, no segundo dia o ED é posicionado no banco de um carro, a uma distância aproximada de 80cm



Fig. 1. Mapa da campanha de medição com destaque para posição do GW.

em relação ao solo, percorrendo os prédios de Educação Física, Medicina e Hospital Universitário, com uma velocidade média de aproximadamente 20km/h. As condições climáticas de ambos os dias estavam favoráveis, isto é, sem chuva e ventos fortes, além de pouca variação de temperatura entre os dias.

A Figura 1 apresenta os pontos registrados durante a campanha de medição, em que os pontos azuis são LOS, em vermelho NLOS e em preto a localização da antena do GW, que pode ser observada de forma mais próxima no quadro em destaque. A identificação de visada é realizada a partir de inspeção visual, com auxílio de um binóculo 60x60 e alcance até 10Km, em cada instante registrado pelo GPS. De forma manual, a identificação de visada de cada amostra é registrada durante a coleta e, posteriormente, inserida no banco de dados.

### III. TÉCNICAS DE CLASSIFICAÇÃO

A campanha de medição resultou um total de 794 amostras, das quais 238 são LOS (29,97%) e 556 NLOS (70,03%). Com isso, podemos aplicar as técnicas de aprendizado de máquina a fim de classificar os sinais coletados. As amostras classificadas como LOS tem valor de visada igual a 1, enquanto as classificadas como NLOS possuem valor igual a 0. Serão aplicadas as técnicas de KNN, MLP, SVM e RF.

As técnicas KNN, MLP e SVM foram escolhidas por serem técnicas clássicas da literatura [16] e, portanto, referências na comparação com outros modelos. Além disso, MLP e RBF SVM são algoritmos robustos contra *outliers*, sendo capazes de compreender relações complexas sem ser afetado por eles. Por fim, a escolha do modelo RF se dá pelo fato de ser um algoritmo de aprendizado de máquina conjunto que combina várias árvores de decisão para criar um modelo assertivo mais robusto contra *outliers*. A combinação de várias árvores de decisão descorrelacionadas permite uma variação de resultados de cada ramo e sua diversificação garante que os erros sejam “sobrepostos” por acertos, fornecendo um modelo mais robusto em termos de desempenho [16].

Antes de iniciar o processo de treinamento, é necessário realizar a extração de parâmetros, a validação cruzada e o pré-processamento, etapas cruciais para aumentar a confiabilidade das técnicas de aprendizado, ampliar a qualidade do treinamento e otimizar o desempenho das técnicas avaliadas.

A extração de parâmetros visa reduzir a complexidade do conjunto de dados, mantendo apenas as características mais relevantes para a classificação, acelerando a convergência da técnica e melhorando a generalização dos dados de entrada. Em outras palavras, alguns parâmetros podem não contribuir

significativamente para a tarefa de classificação, enquanto outras podem ser altamente correlacionadas. Portanto, a remoção de características irrelevantes ou redundantes reduz a complexidade do modelo e evita *overfitting*. Uma forma comum de avaliar a redundância de característica é através da avaliação da matriz de correlação linear entre os parâmetros [16], enquanto a remoção de características irrelevantes pode ser realizada utilizando a seleção de parâmetros baseada em relevância como critério. Este trabalho adotará ambas abordagens para efetuar a extração de parâmetros.

Já o pré-processamento trata os dados antes de serem inseridos nos modelos de aprendizado de máquina, visando identificar e tratar valores ausentes, e, principalmente, balancear os dados. Dados balanceados minimizam valores *outliers* e facilitam o entendimento dos modelos para a classificação, acelerando a convergência e melhorando os resultados finais. Neste artigo, o balanceamento de dados de entrada utilizado foi a normalização, ou seja, todos os parâmetros pertencem ao intervalo entre 0 e 1. 0 o valor mínimo e 1 o valor máximo do parâmetro em questão avaliado.

#### A. Extração de Parâmetros

Primeiramente é necessário selecionar os parâmetros de entrada com base na relevância. O foco desta etapa está na seleção dos parâmetros que possuem relação direta com o problema em análise, garantindo que apenas as informações mais valiosas sejam consideradas para a classificação. Portanto, de imediato são eliminados os parâmetros ‘id’, ‘Time’, ‘GPS Time’, ‘GPS Precision’ e ‘# Satellites’, uma vez que o identificador da amostra e marcadores temporais podem indicar correlações inexistentes e grau de precisão do GPS e número de satélites são irrelevantes, já que todas as amostras compartilham o mesmo grau de precisão e número de satélites.

Seguindo adiante, temos a matriz de correlação linear entre os parâmetros apresentada na Figura 2. É possível identificar, nessa figura, que os pares ‘ED RSSI’ e ‘GW RSSI’; ‘Longitude e Altitude’; ‘Longitude e Latitude’; ‘ED RSSI’ e ‘Altitude’; ‘GW RSSI’ e ‘Altitude’ possuem, em módulo, alta correlação linear entre si. Entretanto, os parâmetros ‘Latitude’, ‘Longitude’ e ‘Altitude’, são imprescindíveis para a classificação correta das amostras em LOS e NLOS, assim, não serão eliminados mesmo que possuam alta correlação com outros parâmetros ou entre si. Com isso, restam os parâmetros ‘ED RSSI’ e ‘GW RSSI’ com maior correlação linear. Como ambos apresentam, em módulo, a mesma correlação entre todos os outros parâmetros avaliados, optou-se por retirar o parâmetro ‘GW RSSI’, pois o objeto de interesse de classificação deste trabalho é a classificação de visada LOS e NLOS de acordo com o comportamento do *End Device*, ou seja, o observado pelo ED. Portanto, o parâmetro ‘ED RSSI’ possui maior relevância, sendo ela adotada como critério de desempate. Desta forma, foram eliminados quatro parâmetros da análise, facilitando a atuação dos métodos de aprendizagem e identificação dos padrões necessários para atingir um bom resultado na classificação. Assim, os parâmetros utilizados como entrada de cada método de classificação são latitude, longitude, altitude e ED RSSI.

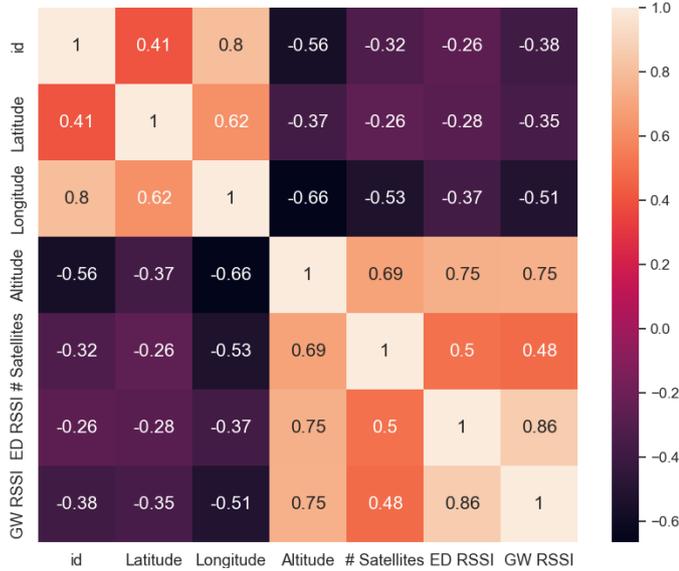


Fig. 2. Matriz de Correlação entre os parâmetros.

### B. Validação Cruzada

A forma mais comum de validação cruzada é através do  $k$ -fold [21], técnica que divide o conjunto de dados em  $k$  subconjuntos (*folds*), onde o modelo é treinado em  $k - 1$  *folds* e validado no *fold* restante. Este processo é repetido  $k$  vezes, garantindo que cada *fold* seja utilizado como conjunto de validação exatamente uma vez. Ao treinar e validar o modelo em diferentes subconjuntos, o  $k$ -fold proporciona uma avaliação mais abrangente do desempenho, minimizando o risco de *overfitting* ou *underfitting*. Isso é especialmente relevante em conjuntos de dados limitados, como neste trabalho, nos quais a variabilidade na distribuição dos dados pode impactar a capacidade do modelo de generalização.

### C. Pré-Processamento

A normalização garante que todos os parâmetros de entrada tenham o mesmo peso durante o treinamento, evitando um viés com tendência para aqueles com maior valor, em módulo, dentro do conjunto de amostras do parâmetro avaliado. Após a transformação, os dados pertencem intervalo ao entre 0 e 1, isto é  $X_{norm} = (X - X_{min}) / (X_{max} - X_{min})$ , em que  $X_{norm}$  é o valor normalizado,  $X$  é a amostra a ser normalizada,  $X_{max}$  é o maior valor do conjunto de dados e  $X_{min}$  é o menor valor do conjunto de dados.

### D. Métricas para avaliação das técnicas

No contexto de aprendizado de máquina, muitas métricas podem ser utilizadas para avaliar a qualidade dos resultados obtidos para cada técnica. Neste trabalho são adotadas as métricas *Score*, *Precision*, *Recall*, *F1 Score* e Tempo de Processamento.

O *Score* é a métrica mais simples para técnicas de classificação [16]. Ele apresenta a relação entre verdadeiros positivos (VP) e verdadeiros negativos (VN) e total de amostras (TA) analisados pelo problema, ou seja,

$$Score = (VP + VN) / TA. \quad (1)$$

Embora simples, essa métrica é muito utilizada em problemas de classificação binária, como neste trabalho, pois permite a visualização da taxa de acerto de forma rápida e intuitiva.

Já o *Recall* é uma métrica que indica a proporção de VP que foram corretamente identificados em relação ao número total de classificações dadas como positivas, ou seja, a soma de VP e falsos negativos (FN) [16], isto é,

$$Recall = VP / (VP + FN). \quad (2)$$

A relevância dessa relação se dá em situações em que FN são considerados mais prejudiciais do que falsos positivos (FP). Portanto, um alto índice de *Recall* implica em um modelo com baixo número de falsos negativos, o que nesse trabalho representa um baixo número de amostras LOS classificadas como NLOS.

A *Precision* é uma métrica relevante [16] para esse tipo de problema, dada sua capacidade de detectar FP presentes na classificação do modelo utilizado. Tal fator é fundamental no contexto deste trabalho, uma vez que classificar um ponto como LOS de forma errônea prejudica o desempenho da rede LoRaWAN, visto que acarreta em uma decisão de redução de recursos por parte da rede, como potência de transmissão e redução de SF. Desta forma, é necessário conhecer a ocorrência FP para dimensionar a porcentagem de pontos que estão susceptíveis a classificação errônea de acordo com cada modelo. Um alto índice de *Precision* apresentado pelo modelo de classificação implica em um modelo com baixa presença de FP, ou seja, uma amostra LOS ser corretamente classificada LOS. Assim, a *Precision* é dada por

$$Precision = VP / (VP + FP). \quad (3)$$

Por fim, o *F1 Score* é uma métrica que combina *Recall* e *Precision*, de tal forma que estabeleça uma média harmônica entre as duas, unindo a detecção de FN e FP [16], conforme

$$F1 = 2 \times (Precision \times Recall) / (Precision + Recall). \quad (4)$$

Um *F1 Score* alto indica um equilíbrio entre a capacidade do modelo de identificar corretamente as amostras NLOS e evitar a classificação incorreta de amostras de NLOS como LOS.

### E. Tempo de processamento

Em aplicações IoT, o tempo de processamento é uma métrica de desempenho importante, uma vez que técnicas complexas de maior carga computacional não são viáveis para classificação LOS/NLOS em aplicações móveis. Além disso, vale ressaltar que grande parte dos dispositivos IoT são limitados em termos de recursos e energia, portanto, é recomendado reduzir os recursos computacionais o máximo possível para aumentar a vida útil da bateria. Por esse motivo, o tempo de processamento para cada técnica é avaliado.

### F. Hiperparâmetros

Todas as métricas de desempenho são avaliadas para cada subconjunto  $k$ , sendo  $k = 10$ , escolha mais comum na literatura [22]. Assim, a média e o desvio padrão dos resultados obtidos nas iterações fornecem uma estimativa mais precisa do desempenho do modelo, reduzindo a sensibilidade a variações específicas do conjunto de treinamento. Por fim, para encontrar os melhores hiperparâmetros para cada técnica, implementamos o algoritmo *GridSearch* com várias combinações de hiperparâmetros distintas. A descrição desse método, bem como dos hiperparâmetros selecionados pode ser encontrada em [17].

TABELA IV

DESEMPENHO DOS MODELOS - MÉDIA E DESVIO-PADRÃO.

	<i>KNN</i>	<i>MLP</i>	<i>SVM</i>	<i>RF</i>
<i>Score (%)</i>	94.08 ± 2.50	95.34 ± 2.95	92.07 ± 2.93	97.48 ± 1.69
<i>Precision (%)</i>	96.04 ± 3.71	91.61 ± 8.08	85.66 ± 7.67	97.28 ± 2.31
<i>Recall (%)</i>	82.03 ± 8.17	92.75 ± 4.41	86.06 ± 7.56	93.47 ± 5.08
<i>F1 Score (%)</i>	88.24 ± 5.57	91.93 ± 4.73	85.57 ± 5.92	95.29 ± 3.46
Tempo (ms)	41.70 ± 5.26	4636.44 ± 1.74	383.60 ± 180.72	61.82 ± 6.69

## IV. RESULTADOS

Os algoritmos desse estudo foram implementados em *Python* 3.11 e executados em um computador com processador i7-8565U (2.7 GHz) e 8 GB de RAM. Mais detalhes sobre a implementação podem ser encontrados em [17]. É obtida a predição da classificação proposta pelos métodos apresentados na Seção III com os hiperparâmetros determinados e em seguida são calculados a média e o desvio padrão para cada uma das métricas de desempenho, que estão apresentados na Tabela IV.

A partir da Tabela IV, é possível afirmar que as técnicas aplicadas tiveram desempenho satisfatório na classificação. Entretanto, conforme destacado na Seção III, é importante observar a classificação errônea, para levar em consideração o peso dos FP e FN, a fim de evitar que a rede LoRaWAN remaneje os recursos computacionais de forma equivocada. Deste modo, o modelo com o melhor desempenho foi o RF, visto que obteve o maior valor para *Score* com um excelente resultado para as métricas *Precision* e *Recall*, resultando no maior valor de *F1 Score* entre todos os modelos. Isso indica uma melhor relação entre baixa presença de FP e FN, sendo esta a razão pela qual o RF se sobressai sobre KNN. Além disso, o desvio padrão de RF foi baixo, principalmente se comparado com os modelos MLP e SVM, indicando que há uma baixa variância na classificação, ou seja, os dados estão sendo classificados de forma próxima em cada um dos *folds* implementados pelo método *k-fold*. Por fim, o tempo de processamento do RF é o segundo mais baixo, mas pouco maior do que o KNN, evidenciando suas capacidades para aplicações IoT.

Embora o desempenho do KNN seja próximo ao do RF, a diferença de desempenho para os outros dois algoritmos é maior. A técnica SVM tem a média mais baixa em todas as métricas de classificação, chegando a quase 10% menor, em média, quando comparada ao RF, com tempo de processamento aproximadamente 6.205 vezes maior do que o do RF. Enquanto isso, o MLP tem uma média geral de classificação mais alta quando comparado ao SVM, mas o tempo de execução é aproximadamente 74.967 vezes maior do que o do RF, desencorajando seu uso em aplicações IoT.

## V. CONCLUSÃO

Neste artigo, é proposto um estudo que compara o desempenho de alguns dos principais modelos clássicos de classificação de aprendizado de máquina presentes na literatura, que são MLP, KNN, SVM e RF. O desempenho dos algoritmos, considerando as métricas avaliadas, foi satisfatório, com destaque para o algoritmo RF, que obteve, em média e desvio padrão, os melhores resultados para o tipo de problema discutido. Além disso, os dados coletados e os programas desenvolvidos para a análise aqui descrita foram disponibilizados.

Como trabalhos futuros, pretende-se discutir uma proposta de implementação de *smart city* para a região do campus da UFJF. Tendo em vista os resultados deste trabalho, é possível realizar testes de soluções que resolvam problemas de tráfego de veículos no campus, tais como semáforos inteligentes e detectores de colisão. Além disso, o consumo de energia de cada algoritmo contribui para a viabilidade de sua aplicação em um cenário de cidade inteligente e para o planejamento da manutenção dos dispositivos. Outra sugestão consiste em aplicar os modelos apresentados neste trabalho para ambientes *indoor*, como fábricas e shoppings, por exemplo, e verificar o desempenho dos algoritmos cenários distintos.

## REFERÊNCIAS

- [1] S. Sinha, "State of iot 2023: Number of connected iot devices growing 16% to 16.7 billion globally," *IoT Analytics* Disponível: <https://iot-analytics.com/numberconnected-iot-devices>, 2023.
- [2] W. Ayoub et al, "Internet of mobile things: Overview of lorawan, dash7, and nb-iot in lpwans standards and supported mobility," *IEEE Comm. Surveys and Tutorials*, vol. 21, pp. 1561–1581, 2019.
- [3] D. Magrin et al, "Performance evaluation of lora networks in a smart city scenario," *IEEE Int. Conf. on Communications (ICC)*, 2017
- [4] V. G. Oliveira e R. R. d. Sabino, "Desafios e soluções para desenvolver a mobilidade urbana sustentável no Brasil," *Zênite Fácil*, 2022.
- [5] E. R. B. Falcão et al, "Prova de conceito de um sistema de monitoramento de fluxo de pessoas baseado em tecnologia LoRaWAN e RFID para o transporte público urbano," *Simp. Bras. de Telecom. e Proc. de Sinais (SBrT)*, 2022.
- [6] D. Magrin, "Network level performances of a LoRa system," *Università degli Studi di Padova*, Dissertação de Mestrado, 2016.
- [7] A. Olejniczak et al, "LOS and NLOS identification in real indoor environment using deep learning approach," *Digital Communications and Networks*, 2023.
- [8] R. E. Nkrow et al, "NLOS Identification and Mitigation for Time-based Indoor Localization Systems: Survey and Future Research Direction," *ACM Computing Surveys*, 2024.
- [9] F. Carpi et al, "Experimental analysis of RSSI-based localization algorithms with NLOS pre-mitigation for IoT applications," *Computer Networks*, 2023.
- [10] J. Yang et al "Orientation Estimation using Differences in WiFi Signal Behavior in LOS and NLOS Cases," *IEEE Int. Conf. on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, 2024.
- [11] Z. Wei et al, "An Intelligent Path Loss Prediction Approach Based on Integrated Sensing and Communications for Future Vehicular Networks," *IEEE Open Journal of the Computer Society*, vol. 5, pp. 170-180, 2024.
- [12] A. Bombino et al, "Machine Learning-Aided Classification Of LoS/NLoS Radio Links In Industrial IoT," *IEEE Int. Conf. on Factory Communication Systems (WFCS)*, 2020.
- [13] J. Finnegan et al, "Modeling the Energy Consumption of LoRaWAN in ns-3 Based on Real World Measurements," *Global Information Infrastructure and Networking Symposium (GIIS)*, 2018.
- [14] R. S. Macêdo et al, "Modelagem Inteligente de Perda de Percurso Utilizando Transceptores LoRa," *Simp. Bras. de Telecom. e Proc. de Sinais (SBrT)*, 2019.
- [15] K. Z. Islam et al, "LoRa-based outdoor localization and tracking using unsupervised symbolization," *Internet of Things*, vol. 25, 2024.
- [16] K. P. Murphy, "Machine Learning: A Probabilistic Perspective," *MIT Press*, 2012.
- [17] L. L. de Oliveira, "Identificação Inteligente LOS/NLOS LoRaWAN UFJF", Disponível: <https://github.com/lucas-lima99>, 2024.
- [18] Multitech, "MTAC-LoRa Cards," Disponível: <https://www.multitech.net/developer/products/multiconnect-conduit-platform/accessory-cards/mtac-lora/>, 2024.
- [19] Multitech, "Multitech Developer Resources - mDot," Disponível: <https://www.multitech.net/developer/products/multiconnect-dot-series/multiconnect-mdot/>, 2024.
- [20] L. A. de Oliveira, "Arcabouços para Coleta de RSSI e Evolução de Técnicas de Acordo de Chaves em Redes LoRaWAN". *Universidade Federal de Juiz de Fora*, Trabalho de Conclusão de Curso, 2023.
- [21] C. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
- [22] I. K. Nti et al, "Performance of Machine Learning Algorithms with Different K Values in K-fold Cross-Validation," *Int. Journal of Inform. Tech. and Computer Science (IJITCS)*, vol. 13, 2021.